AFRL-IF-RS-TR-2005-1
Final Technical Report
January 2005

COMMAND POST OF THE FUTURE

Global Info Tek, Incorporated

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. J498

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

STINFO FINAL REPORT

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2005-1 has been reviewed and is approved for publication

APPROVED: /s/

MARK D. FORESTI
Project Engineer

FOR THE DIRECTOR: /s/

JOSEPH CAMERA, Chief
Information & Intelligence Exploitation Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	JANUARY 2005	Final Sep 99 – Mar 04

4. TITLE AND SUBTITLE
COMMAND POST OF THE FUTURE

5. FUNDING NUMBERS
C - F30602-99-C-0263
PE - 63760E
PR - CPOF
TA - 00
WU - 06

6. AUTHOR(S)
Ray Emami

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Global InfoTek, Incorporated
156 East Maple Avenue
Vienna Virginia 22180

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Defense Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington Virginia 22203-1714

AFRL/IFTB
525 Brooks Road
Rome New York 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
AFRL-IF-RS-TR-2005-1

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: Mark D. Foresti/IFTB/(315) 330-2233/ Mark.Foresti@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*
This report represents Global InfoTek, Inc., (GITI) efforts in support of their piece of the DARPA sponsored CPOF program. GITI's role in supporting the CPOF Program has been to provide the following support services: Systems Engineering and Integration, Experiment support. Modeling and Simulation, Hardware Allocation and Inventory Control, in addition to overall support and coordination of activities.

14. SUBJECT TERMS
Situation Awareness, Battlefield Visualization, Automated Reasoning, Plug-n-Play Architectures

15. NUMBER OF PAGES
44

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

Table of Contents

List of Figures

1. Introduction

The goal of the Command Post of the Future Program (CPOF) is to shorten the commander's decision cycle to stay ahead of the adversary's ability to react. To accomplish this goal, CPOF has developed software that supports the following objectives:

- Increase Speed and Quality of Command Decisions
  - Faster recognition and better understanding of the changing battlefield situation
  - Faster and more complete exploration of available courses of action (COA)

- More Effective Dissemination of Commands
  - COA capture for dissemination of commander's intent
  - Status and capability feedback from deployed operators

- Smaller, More Mobile and Agile Command Structure
  - Fewer staff members forward
  - More mobile, distributed command elements
  - Smaller support tail and reduced deployment requirements

Large, highly staffed command centers are vulnerable to attack. The CPOF program is developing business processes, interactive visualizations, and collaborative decision support tools that will:

- Make the command post commander-centric rather than place-centric
- Eliminate fixed command posts from tactical areas of operation
- Allow distributed, collaborative operation from anywhere in the battle space
- Improve situational analysis, planning and execution
- Replace the need for face-to-face interaction with advanced technology
- Supply users with intuitive interfaces that can be recomposed based on need

The pages that follow in this report represents Global InfoTek, Inc. (GITI) efforts in support of their piece of the DARPA sponsored CPOF program. Global InfoTek, Inc. (GITI) has been supporting DARPA on Command Post of the Future (CPOF) program for several years. The Program has taken on many different faces and directions in research. In support of this effort, GITI has proven the ability to be ever changing in the roles we have played and the services our team members have provided. Global InfoTek, Inc. has made every effort to be proactive to the needs of our DARPA PM. We have followed in the tradition of our company to provide creative "out of the box" solutions to DARPA hard technical issues.

Our role throughout the CPOF Program has been to provide the following support services: Systems Engineering and Integration, Experiment support, Modeling and Simulation, Hardware allocation and Inventory Control in addition

to overall support and coordination activities. Global InfoTek, Inc. has received high accolades over the course of the program for our efforts in the form of letters of recommendation and, most recently, with the award of the 2004 DARPATECH Significant Technical Achievement Award.

In the advancement of our program initiatives, there have been many successes and equal opportunities to learn from our experience. These efforts should hopefully serve as models of success for future DARPA programs and should change the future of Command and Control. Lessons learned on CPOF will provide the future Army with a transition effort of the technology that "is changing the way we do business" – MG Peter Chiarelli 1CD Commanding General. Another organization has been tasked by DARPA to capture and analyze the lessons learned. In this report, we have included the lessons learned with supporting CPOF Systems Engineering and Integration task.

2. System Engineering and Integration Framework

The Systems Engineering team at Global InfoTek (GITI) provided expertise in the development of a baseline system architecture that supported CPOF experiments, and the establishment and technical leadership of CPOF working groups. A system engineering web site and CPOF email lists have been provided to support communication and collaboration within the program and to organize CPOF system engineering related technical specifications and interface definition documents.

2.1 Baseline Systems Architecture

GITI provided team leadership and guidance for the technology providers in support of the DARPA PM's vision and goals. This activity included the creation and dissemination of a system architecture development plan that identified system integration objectives, plans to reach those objectives, and a systems architecture that will support the experiment requirements of CPOF. The plan included the development of an organizational infrastructure based on working groups to resolve technical issues and provide a forum for focused discussion and coordination among technology providers. The CPOF architecture consists of a set of "integration points" aimed at providing a common set of interfaces for inter-system communication while enabling flexibility of communication between components within a subsystem.

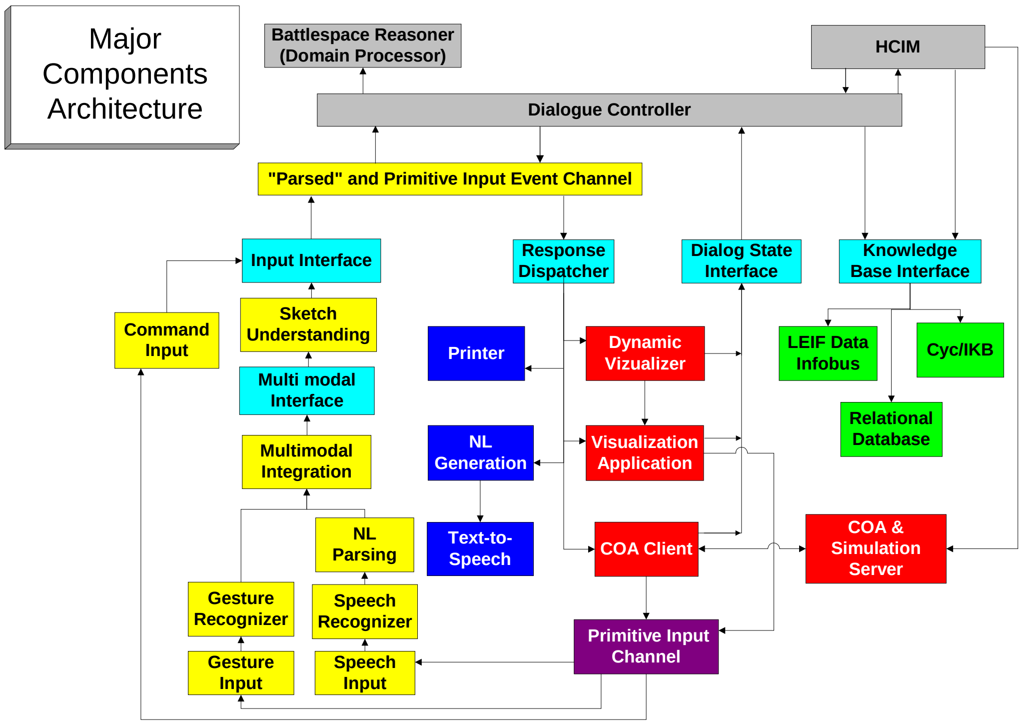


Figure 2.1 – Baseline CPOF Architecture

GITI's system engineering team developed an overall integration strategy and system architecture for integrating NGII, BADD, DDB, and CPOF visualization in support of the Multi-sensor Tactical data Visualization (MTV) TIE.

The systems engineering team identified the long-term value of considering the use of DARPA’s Control of Agent Based Systems (CoABS) program’s Grid in order to integrate the CPOF subsystems in loosely coupled federated systems architecture.

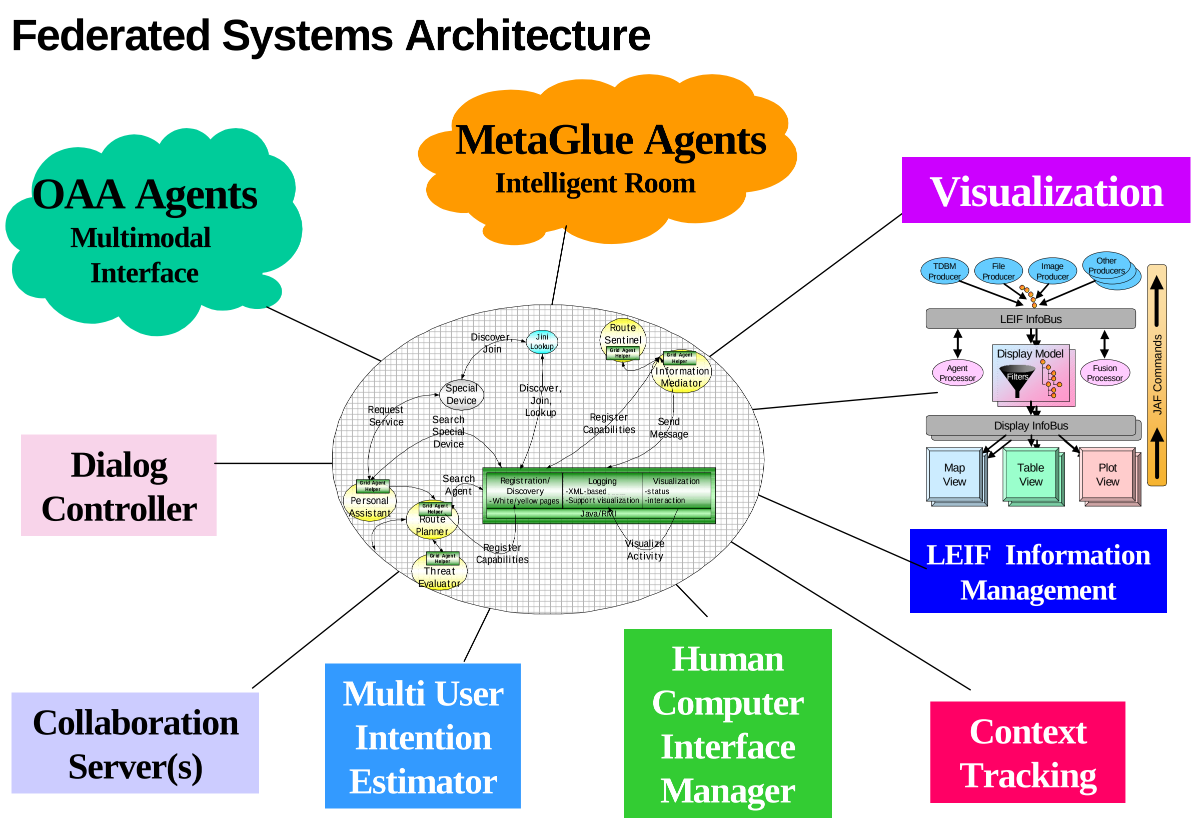


Figure 2.2 – Federated Systems Architecture

2.2 Working Group Technical Leadership

The systems engineering team identified the need for sub architecture mini-workshops and coordinated activities with key personnel from the Visualization, Multi-Modal, Dialog and Context Management, Information and Knowledge Management, and Course of Action (COA) working groups to establish and host workshops to identify and resolve hard integration issues. The systems engineering team worked closely with each group to evaluate alternative subsystem architectures. The systems engineering team identified possible duplication of capabilities between various working groups and helped to specify the revised system integration as each of the groups fleshed out more of the detailed design.

For example, the ***Visualization working group*** focused on four objectives: a visualization toolkit for a virtual common canvas, a workspace with pervasive

user interface “physics”, coordination with Multi-Modal and Dialogue Management work group, and customized and automated visualization. MAYAVIZ developed a downloadable demo of the CPOF map and proposed an API for integration with CMU Multi-Modal. VDI distributed a first cut of a visualization application that integrated a navigable 3D-terrain environment with an interactive data display. Their focus was to create an animated “blobology” algorithm that worked at several echelons and also had terrain-sensing abilities.

The ***Multi-Modal working group*** focused on several APIs including an API for Ink and Gesture.

The ***Dialog and Context Tracking working group*** published an architectural framework for integrating dialog and context tracking with other CPOF components as well as an API for dialog management.

The **Domain Model working group** focused on semantic interoperability, common knowledge representation, component-based information sharing, and re-use of components, coordinate limited objective experiment data gathering and domain modeling by identifying common data sets. Their objective was to consolidate the requirements, provide consistent data to all developers, leverage existing sample data sets to facilitate rapid analysis of the architecture, streamlined integration, and shared development resources.

The ***COA working group*** was focused on supporting COA *generation* and COA *comprehension*. Some of the issues under discussion regarding COA generation include understanding the enemy, the playing field, blue capabilities, opportunities for action, dynamics of flow over time, and defeat mechanisms. COA comprehension entails communicating/understanding the assumptions/factors that entered into generating the COA. This includes the ability to repeat back a COA that was told (i.e., to ‘brief back’), understanding the intent of the COA as well as the specific tasks to be accomplished, and understanding the dynamics of how the COA would flow over time.

3. Experiment Support – Overview

Global InfoTek, Inc. supported many CPOF experiments and data collection. Our overall charter as specified by the program manager was to support objectives across the CPOF program, across DARPA programs, and with other military organizations. Our support for experiments spanned a variety of activities to ensure successful experiments and data collection. GITI participated in data reduction, analysis of the results, and distribution of experiment data to technology providers. This task required GITI to organize and lead many technical coordination activities. Our technical support for experiments included the following activities:

- Provided leadership and coordinated activities of all CPOF participants to make sure everyone is fully cognizant of the Program Manager's objectives for the experiment. This involved organizing weekly conference calls and meetings and ensuring all activities are coordinated for supporting the experiments. Our activities resulted in many informal and formal successful experiments

- Supported with weekly Electronic Tactical Decision Games (eTDG) and all of the CPOF Block Parties. We have received numerous accolades from the Program Manager for the management and support of complex experiments and collecting valuable multimedia technical data. Collected data was delivered to the Program Manager and distributed to technology providers.

- Supported the Program Manager in numerous technical presentations and technology demonstrations at DARPA and military organizations.

- Supported all of the CPOF Subject Matter Experts (SMEs) in their requirements for software and hardware by providing technical trouble shooting, training, and infrastructure support to the SMEs.

3.1 Limited Objective Experiment

GITI installed infrastructure hardware, Jumpstart software, and new CPOF unique software applications at DARPA's Technology Integration Center (TIC). GITI coordinated activities with the TIC security manager to receive SIPRNET accreditation and connection to Intelink-S. This effort required reconfiguration of the systems to accommodate security requirements. CPOF received its accreditation in late June 1999.

GITI staff coordinated with the user community to receive and install operational C4I systems at DARPA's TIC including the Marine Corp's IMMACS and LEIF data producers from the BADD program. GITI installed and configured JSAF Modeling and Simulation system at DARPA's TIC.

GITI staff, in collaboration with Evidence Based Research, Inc designed the infrastructure to support the LOE. GITI provided technical details for key

technology areas, identified technologies that can be used to support experimentation, and provided technical feedback on EBR's "CPOF Model".

GITI selected and procured digital video cameras and a digital video editing station to capture video at meetings to support the LOE subject documentation efforts. This contributed to the design of LOE1. GITI participated in the operational support of LOE1. GITI provided equipment and logistic support for the computational environment and audio/video taping efforts for the LOE1 pilot and actual LOE1 event.

3.2 Electronic Tactical Decision Games (eTDG)

GITI was tasked by the Program Manager in December 1999 to evaluate Web based collaboration products. The goal of this evaluation was to identify a specific product which can be used to replace or complement collaboration tools that supported eTDGs. The eTDGs were organized by Subject Matter Experts to teach technology providers the arts and sciences of military decision making in a collaborative environment. The technology in December 1999 that was used to support the eTDG was based on Shockwave technology.

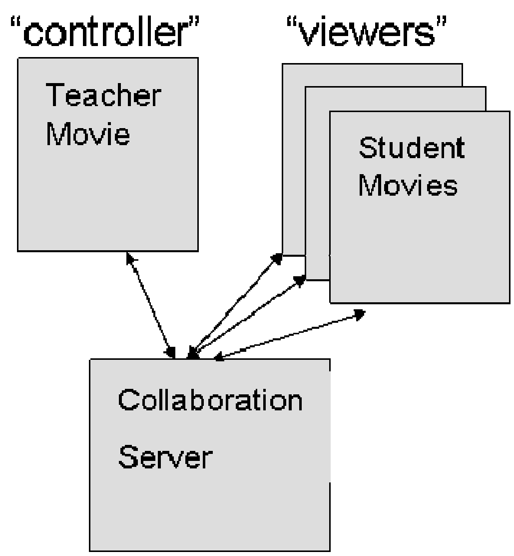


Figure 3.1 – eTDG Diagram

Key features of this technology were:

- Ability to turn on/off object visibility
  - Instant updates of student maps (e.g. teacher loads explosion layer)
- Lingo script language
  - Well documented API
  - Can communicate with VB and JavaScript
- Ability to manipulate objects
- Dynamic support for external objects such as PPT, GIFs, etc.

Limitations of the Shockwave architecture that necessitated search for an alternative technology included:

- Lingo is yet another language to learn
- Platform support currently limited to Windows and Mac
- Third party solution required for video streaming
- Scalability issue
    - Private features (whiteboard layers, chat) linked to user license connection

3.2.1 Requirements for enhanced eTDG

The requirements for the collaborative products evaluation were derived from participating and observing several eTDG events. The technical requirements to support eTDGs were:

- “Must Have” Requirements
    - Individual object manipulation (move, erase, etc.)
    - Dynamic image sharing
    - Ability to host a maximum of 50 simultaneous licensed connections
    - Dynamic briefing and presentation capabilities
    - Reduce the time needed to create a new eTDG’s
- Desirable Requirements
    - Firewall tunnel
    - Portable across Macs, PC’s, and UNIX
    - Dynamic creation of private sessions (e.g. chat rooms, whiteboards, etc)
    - Voice over IP (VoIP) support
    - Streaming multimedia support
    - One time software installation and download
    - Ability to move mouse over objects as a way to provide amplifying information

3.2.2 Candidate eTDG Products Evaluated

GITI evaluated several alternative leading technologies with use potential to support eTDG. The candidate products considered were:

- MITRE’s Collaborative Visual Workspace
- Microsoft NetMeeting 3
- JDH Technologies Web-4M
- Marratech
- Thoughtstar QuickTeam Professional
- TeamWave 4.3
- Visual Rendezvous 2.7
- NetManage eDemo
- PlaceWare Conference Center 3.5

- Contigo i2i Meeting Pro
- Astound 6.0
- Instinctive eRoom 4.1
- Centra 99
- WebEx
- Tango 1.4

Based on our analysis and consultation with the Program Manager, GITI recommended Web-4M as the best candidate to support eTDGs. Key features of the Web-4M are:

- Chat Room based metaphor
  - Server administrator creates permanent rooms
  - Users create local hideouts and annexes, which are deleted when the last user leaves these areas
  - Individuals can be in multiple rooms
- Synchronous collaboration
  - Instant Messaging
    - Instant messaging between users and groups
    - Users and groups can instant message in permanent and temporary rooms
  - Chat
    - Chat users can be identified by color
    - Private chat available between one or multiple users
  - Whiteboard
    - Supported inside Slide Show feature
    - Object based metaphor
    - Ability to control whiteboard modifications or modifications to objects drawn on the whiteboard
  - Audio Conferencing
    - Integrated with Web-4M
    - No reliance on 3rd party applications
  - Slide Show
    - Supports whiteboard images, JPEG's and GIF's, URL's, Questions
    - Ability to control annotations
    - Supports mini-chat feature

3.2.3 Enhanced eTDG Deployment

GITI developed a collaborative eTDG based on Web-4M product. Technology providers and SMEs were logging into a GITI provided server and participated in the eTDG. This technology enabled highly interactive and dynamic interactions among participants.

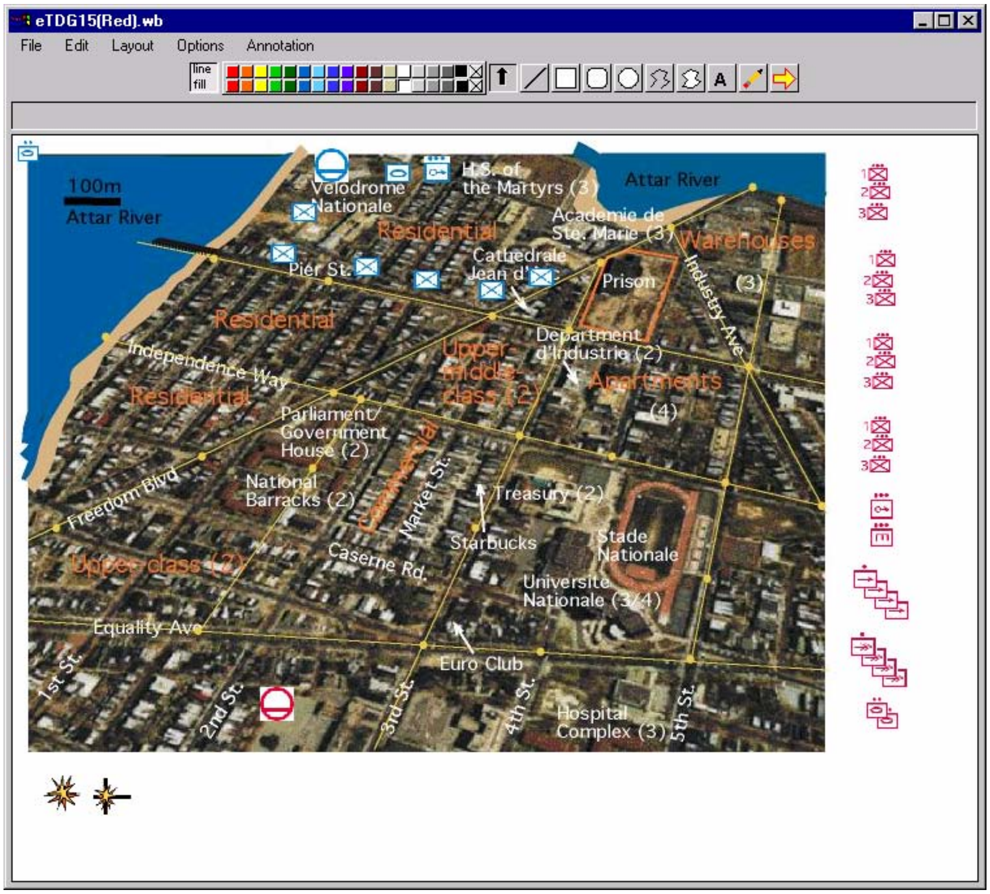


Figure 3.2 – WEB 4M Screenshot

A GITI-configured Web-4m server was used for three presentations during the Urban Warfare Workshop to eight participants who were connected across the country via the Internet. Web-4M received positive feedback from the on-line participants. Web-4M was also used for presentations during Block Party III to six participants across the Internet.

3.3 Block Parties

These "Parties" were a week long series of command experiments to test blocks of technologies in exercises. The most promising technologies were selected for further development within the program.

3.3.1 Block Party III (Urban Operations) – Experiment

GITI designed, developed, integrated, and configured the Block Party III testbed and the entire data collection facility. The testbed design included computer hardware, supporting software for data collections, network, audio recording and distribution from multiple stations and capturing multiple time synchronized video from many angles to ensure capture of all activities. GITI participated in the Visualization Working Group meeting in Pittsburgh 20-21 January 2000. In

support of the Program Manager, GITI coordinated the purchase of urban model equipment, including scenery and models, for construction of the Hue City model at IDA. The Hue City physical model was used for Tactical Decision Games (TDGs) at the Urban Operations Workshop (during early March) and Block Party III (during late March).

Figure 3.3 – Block Party Diagram

GITI provided a videographer to support the Urban Operations Workshop and CPOF Block Party held at IDA. These video tapes were then converted from miniDV format to VHS and provided to all of the CPOF contractors for analysis. The objective of this exercise was to use an urban model to visualize troop management in urban terrain. Goals met for this exercise were Education of the developers in the CPOF double helix development cycle to observe 3D visualization of soldiers in urban terrain to create future software design principles for a command and control system. One of the lessons learned from this activity was that our group needed to continue its development approach to provide a greater level of cross experience sharing between the technical folks and Subject Matter Experts.

3.3.2 Block Party IV – Experiment

GITI designed, developed, integrated, and configured the Block Party IV testbed and the entire data collection facility. The testbed design included computer

hardware, supporting software for data collections, network, audio recording and distribution from many stations, capturing multiple time synchronized video from many angles to ensure capture of all activities. We found in this exercise there was a great deal to be learned in the collection of DATA. We learned lessons that some of the sensors and technology we were trying to implement were not as ripe and mature as was needed to continue its use in the program. Although not all our goals and objectives of data collection were met; we learned that there was a common trend of using an open data structure for the collection and holding of information.

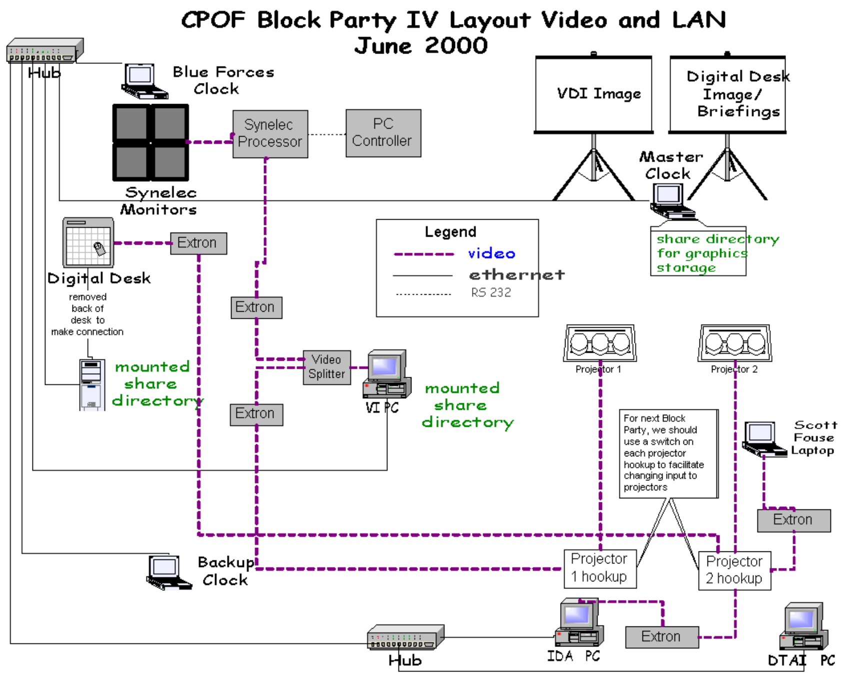


Figure 3.4 – Video and Local Area Network Diagram

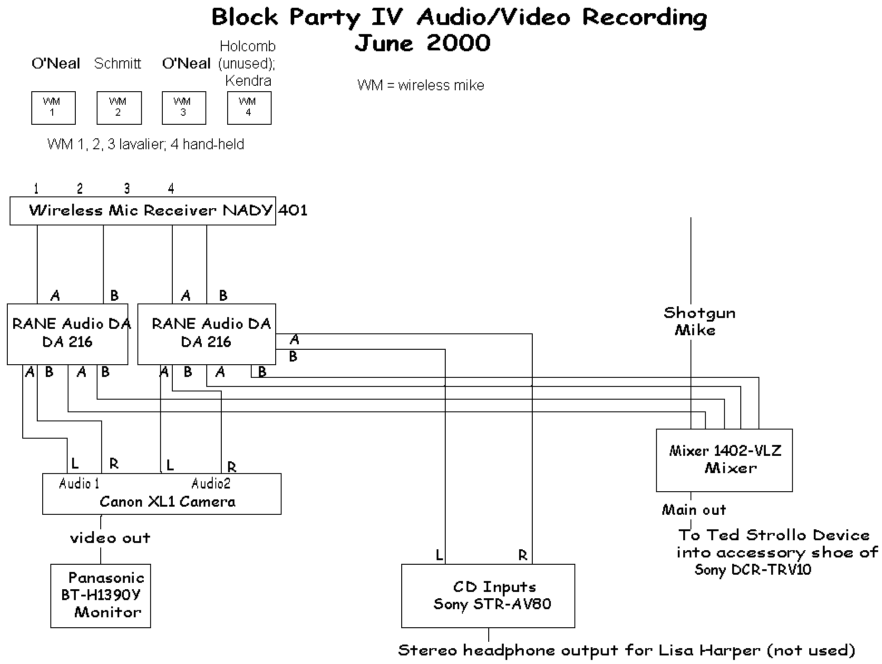


Figure 3.5 – Video and Video Recording Diagram

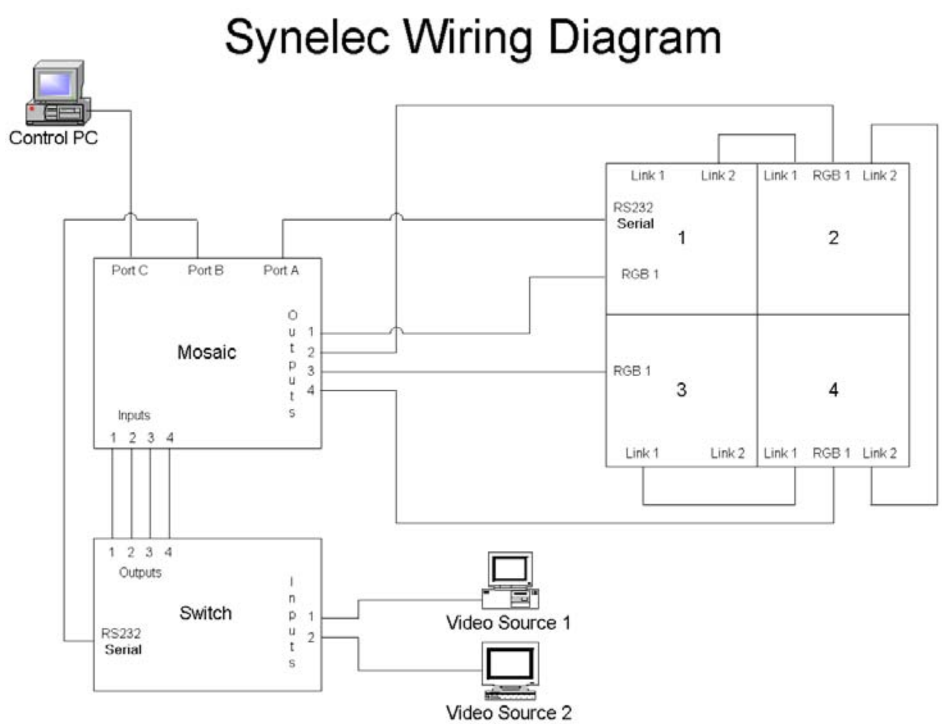


Figure 3.6 - Synelec Large Screen Display Diagram

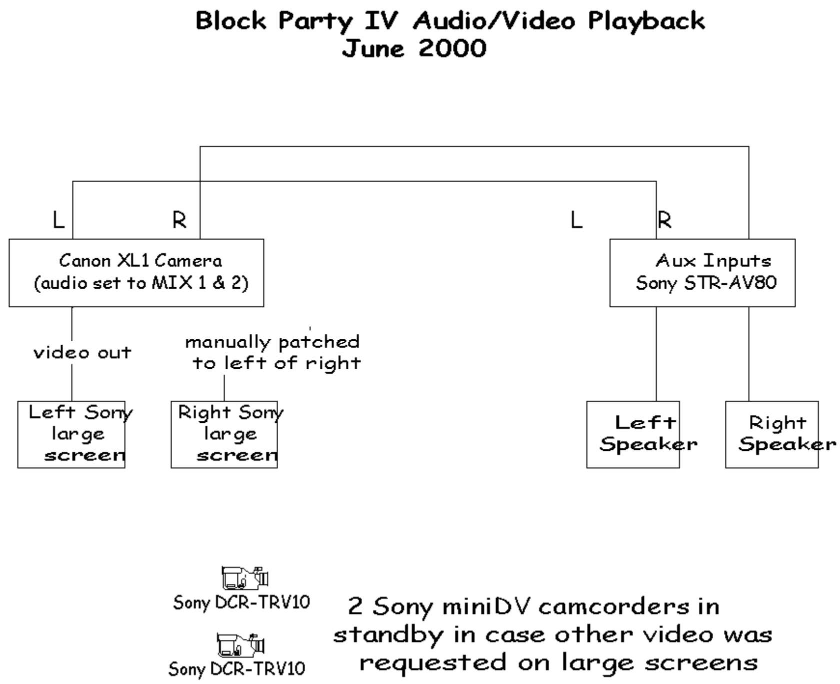


Figure 3.7 – Audio/Video Playback

3.3.3 Block Party V – Experiment

GITI designed, developed, integrated, and configured the Block Party V testbed and the entire data collection framework. The test bed design included computer hardware, supporting software for data collections, network, audio recording and distribution from multiple stations, capturing multiple time synchronized video from many angles to ensure capture of all activities.

The technical details for this Block Party V testbed experiments are provided in the following diagrams. For this exercise it was the objective to further our endeavors in data collection and storage. A goal of this exercise was to collect and store data and provide presentation of this data in many ways. We tried to provide numerous visualizations of the data in Synelec and Digital Desk. We learned from this exercise the concept of creating a bit bucket of data to which each visualization created its own picture based on each mans experience and knowledge.

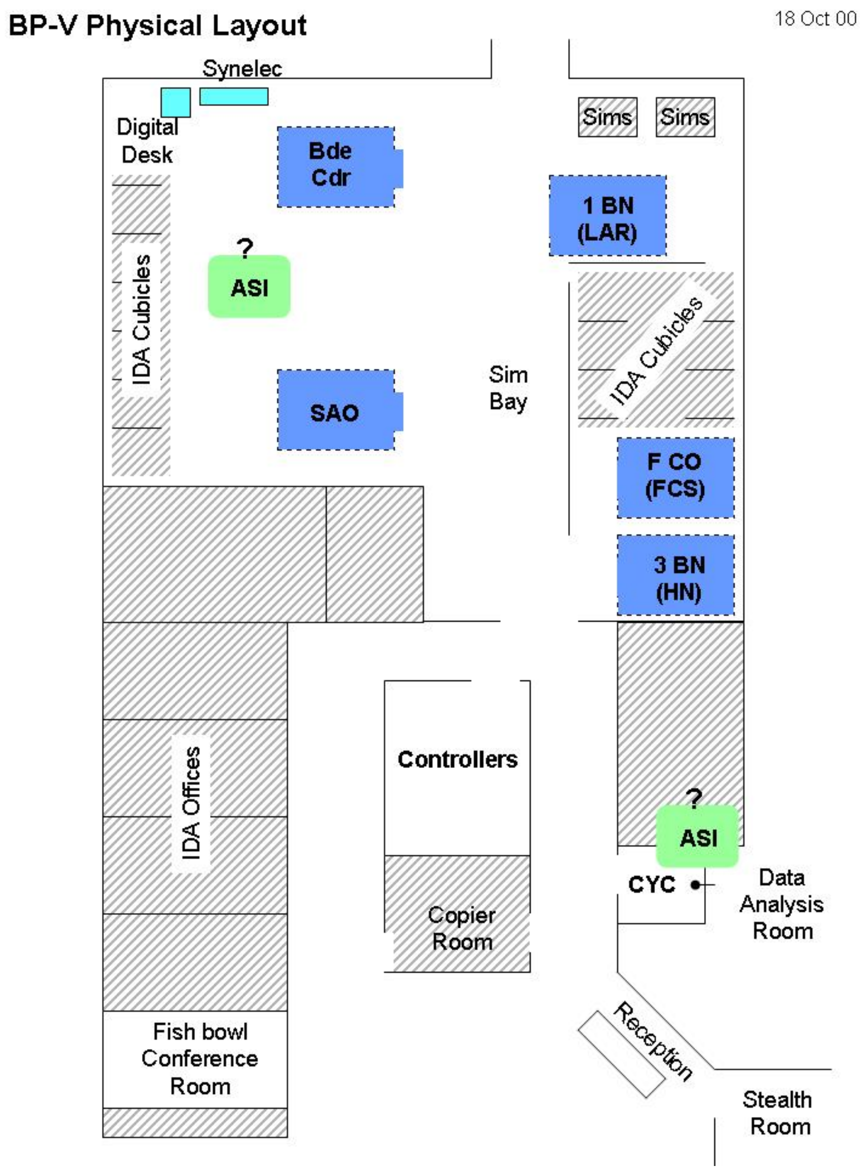


Figure 3.8 – Block Party V Physical Layout

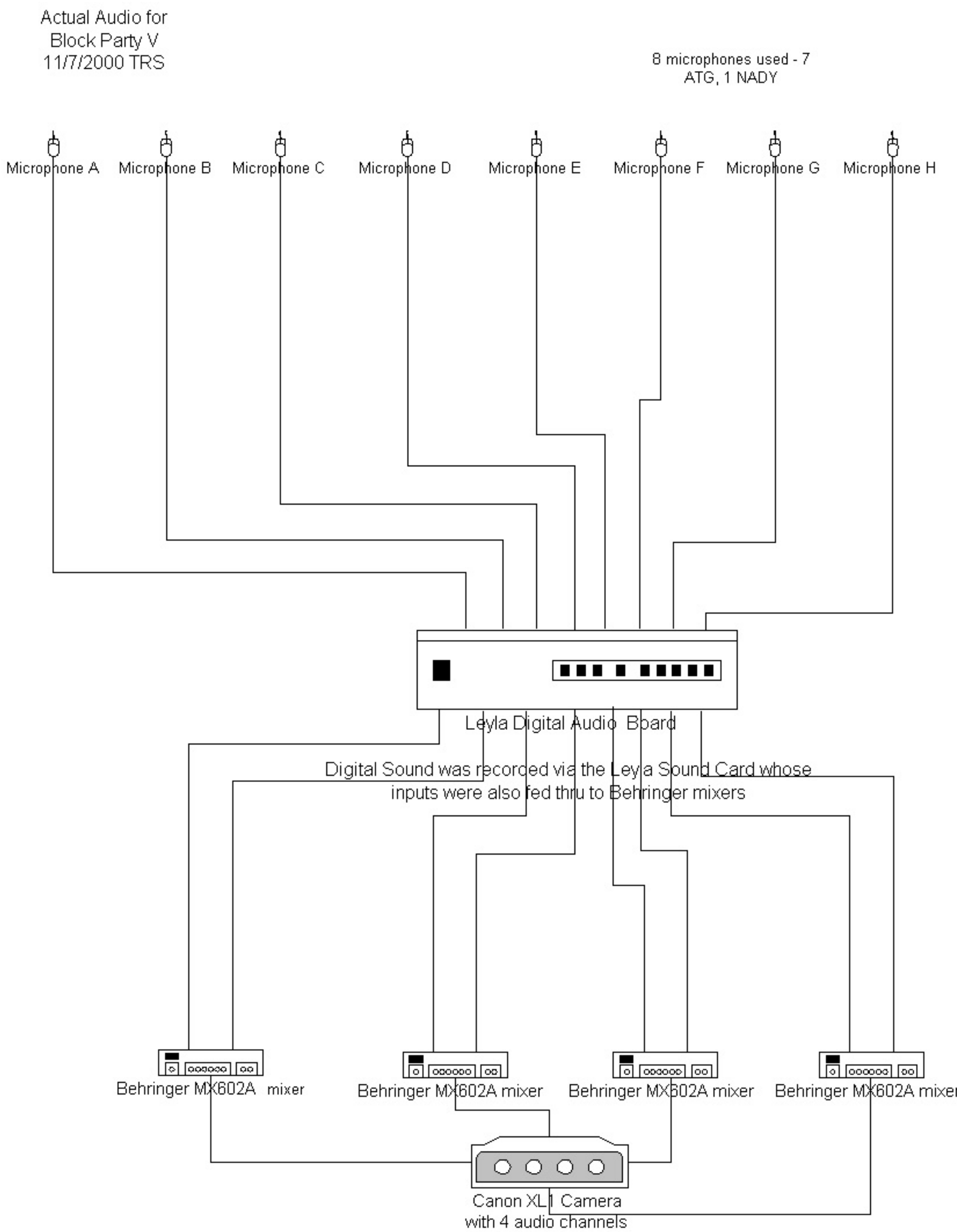


Figure 3.9 - Block Part Sound Wiring Diagram


BDE Commander


1st BN


Audio Table


F Co (FCS)


SAO


3rd BN


Controllers


Data Analysis Room

Figure 3.10 – Images from Block Party

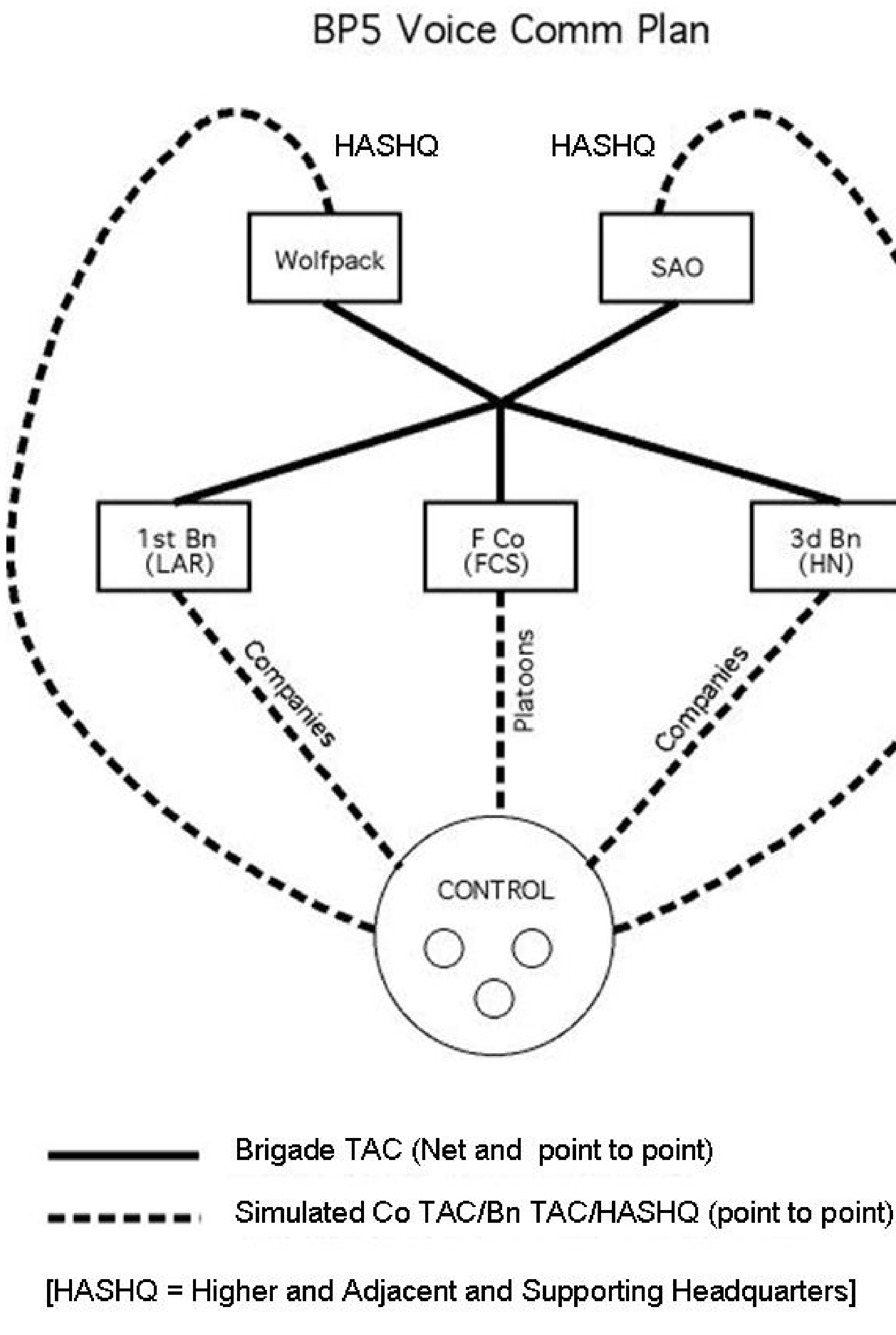


Figure 3.11 – Block Party Comm Plan

GITI personnel attended a Fort Knox, KY recon meeting September 20 – 22, 2000. We worked diligently with the relevant people to establish network and data connectivity between IDA and Fort Knox for future CPOF efforts. At the request of the Program Manager, on August 7, 2000 we removed the equipment which was in the CPOF lab at the TIC (the ALP project took over this lab and CPOF now uses IDA instead). The equipment is currently stored at GITI and is used for CPOF purposes both at GITI and IDA.

3.4 Collaboration Experiments (Collab Ex)

The Collab Ex series refined the technologies which emerged from the block parties and focused them more on a team based cognition through the use of collaborative command and advanced visualization techniques for rapid situation understanding. The following was GITI's role and responsibility in this series of experimentation.

3.4.1 CollabEx 1

GITI participated in and supported CollabEx1 at MAYAViz March 13-19, 2001. GITI staff recorded 75 hours of audio and video data including the actual exercises and the hot washes before and after the exercises all at the direction of Ward Page, the Program Manager. With the adoption of our path to develop a software based Command and Control tool to be used by commanders, this was our first opportunity to work our established double Helix development cycle. Our goal of this experiment was to join Developers and Greybeards in a game to decide development lifecycles and collaborative ideas. We leaned that we had an opportunity to be reactive to requests and began to work like a virtual team of many company's and players.

3.4.2 CollabEx 2

GITI was responsible for hosting an exercise called CollabEx 2. The actual exercise took place Dec 10-14, 2001 at GITI. A dry run was also held at GITI prior to the actual exercise. GITI designed, developed, integrated, and configured the testbed and the entire data collection facility. The testbed design included computer hardware, supporting software for data collections, network, audio recording and distribution from multiple stations, capturing multiple time synchronized video from many angles to ensure capture of all activities. It was our objective of this exercise to work out the wrinkles in playing a full scale simulation against our software using multiple players in our collaborative environment. During this exercise we began to explore voice collaboration tools that could be used for effective communication within the CPOF system. We learned that we would need to continue to provide large amounts of support roles during the exercises from logistics to network operations and monitoring.

3.4.3 CollabEx 3

GITI hosted CollabEx 3 on February 19-21, 2002. GITI designed, developed, integrated, and configured the Block Party V testbed and the entire data

collection facility. The testbed design included computer hardware, supporting software for data collections, network, audio recording and distribution from multiple stations, capturing multiple time synchronized video from many angles to ensure capture of all activities. For this experiment, GITI designed and implemented a custom communication system for exercise participants using 8 telephones with 2 lines each. GITI captured 18 hours of the exercise on digital video. One of the objectives of this event was to provide voice to the players in the collaboration. GITI played a huge role in this undertaking. Early solutions that were not effective in providing a full duplex solution were the use of Motorola talk radios in the communication system. We learned to try many solutions to the problems until we found the best one. We also continued to work on better more deployable solutions for long term rollout. Another goal of this exercise was to continue the double Helix development, which included playing sessions from the SME's and discussions with the developers and designers.

3.4.4 CollabEx 4

GITI planned and developed the testbed for CollabEx 4 in April, 2002. For this CollabEX GITI investigated the use of alternative communication solutions for CPOF exercises. GITI identified ASTi technology for approximately $66,000 w/90 day lead time. Our team designed a more cost effective, GITI-developed solution that included 9 intercom loops, with 2 switch boxes and a phone at each station. This allowed us to provide 8 separate conference discussions between experiment participants and a Broadcast channel. The voice solution provided full duplex sound at phone quality. The objectives in this exercise were to continue our double helix development and to fully integrate our voice solution into the rhythm of the simulation. We learned that there were some additional technical difficulties to be worked out in our voice solution but had to deal with these issues while continuing with the integration of technology. We met a further opportunity to challenge our developer's with new features and bug fixes. In the following diagram below you can see the design specifications for one conference loop control:

```
-----------------
|                |        50 ohm 10W              120V:12V 3A
|  12V DC 1.2A   |+ -------/\/\/\/\----+-----------
|  Power Supply  |                     |+         ) # (--
|                |          4700 uf -----         ) # (
|                |             35V  /---\         ) # (--
|                |                     |          ) # (
|                |- -------------------+         _) # (__
|________________|                     |         |
                                       |         |
                                       |         |
                                       ^         ^
                                      Red      Green

                                    To Telephone Jacks
```

Figure 3.12 – GITI Analog Comm Wiring Diagram

Figure 3.13 – 8 Loop GITI Analog Controls Wired on Board

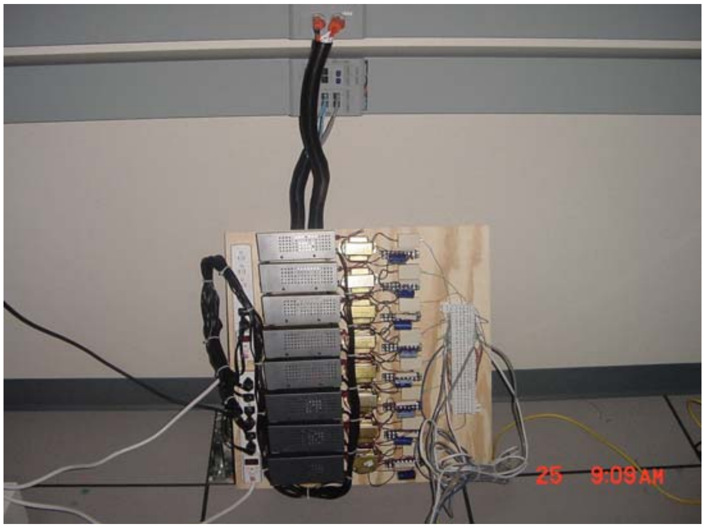


Figure 3.14 - Loop Control Units Wired to Stations

3.4.5 CollabEX 5

GITI staged and supported three additional separate CollabEx sessions at ISX in Arlington, VA between April and June 2002. These session were recommended by Ward Page, the DARPA Program Manager, to determine the stress level of each of the technologies being offered at this level of the experimentation process. These sessions increased in complexity and the number of workstations each time, culminating in a session with six players and four controllers. Each participant was provided a computer workstation with multiple displays. GITI designed, implemented, and tested the hardware communications system for these exercises. The system consisted of 20 telephones at 10 workstations

operating on 9 separate party lines. The setup includes custom power supplies and dedicated wiring to each station. Each workstation supported up to four sets of headphones simultaneously. This solution allows for voice communications between exercise participants as well as silent monitoring for guests who are trying to listen into the exercise itself.

GITI supported each of the CollabEx sessions with network and system administration expertise. GITI was instrumental in providing a stable network during the exercises at ISX. Several network issues were avoided during the course of the program activities, and GITI was able to support network connectivity for printing and wireless access. GITI also served as the system administrator for all the exercise equipment, including network configuration, virus protection, operating system patch installation, software license management, and hardware.

GITI developed a custom messaging system for communications coordination during CPOF exercises. At Ward Page's request, GITI designed and implemented a custom computer messaging solution to serve as a backup coordination mechanism during these exercises. The software was developed in Java and ran in a client-server environment. It was able to support virtually unlimited users, including the same user logged in more than once.

GITI managed the CPOF program audio-visual capture efforts, including camera and audio work during the exercises. Approximately 70 hours of high-quality video were shot with audio from the communications system being recorded simultaneously. At the last exercise, there were three cameras running at all times, with each connected to the subject's communications setup and a wireless microphone system to capture ambient conversations. The objectives of this event were threefold.

- Further the understanding of the program developers in how the system is used and give them better development ideas based on these understanding
- Prepare the application through collaborative player sessions
- Learn support function expertise and develop testing environment

I feel we were successful in accomplishing these goals. We learned we had built a system to provide a greater understanding of our CPOF abilities and environments.

3.5 Command Experiments (Command Ex)

As the technologies matured, it became apparent that conducting experimentation and evaluating how these technologies performed within the nature of tactical command was critical to the success of the program. The CommandEx is a one-week exercise staffed by retired and active duty officers, whereby, the participants collaboratively build the enemy story from salute

reports and situation reports to gain situational awareness using the CPOF technologies.

3.5.1 CommandEx 1

Originally called ControlEx; Global InfoTek, Inc. supported this event held at MAYAViz in Pittsburgh PA. This event was held in September 2002. The event was the first opportunity for the GITI team to do performance analysis at the event between a single Processor P4 system and a Dual Xeon processor system. At this event, GITI increased the memory of our baseline machine to 2GB of Rambus memory. The Goal of this exercise was to create a larger group of controllers and play the CPOF system in a Collaborative session with 2 types of enemy. An urban gorilla and a cold war type aggressor were used in the simulation. This session created a hard test for the CPOF system and showed how slow the system performed.

3.5.2 CommandEx 2

GITI supported this event in November 2002 held at ISX in Arlington VA. GITI began replacing the baseline player machines from the older 1.9GHZ Micron machines to the new Dual Xeon based Compaq workstation machines. GITI extended the network cabling to include additional Player stations. Our goal in this exercise was to increase the hardware requirements to see if we could alleviate some of the system slowdowns and sluggishness. We increased the system hardware to meet this objective. Our testing showed improvements and signs that further HW increases could bring us more system speed.

3.5.3 CommandEx 3

GITI supported this event held at ISX in February 2003. The new addition to the Experiment environment was a PBX phone system. Global InfoTek, Inc. worked with engineers from ISX to procure a Windows Server based PBX, consisting of conference bridge cards. GITI worked with two outside vendors in the setup and installation of this voice system. In conjunction with the installation of this PBX, GITI continued to setup our manual voice system. Both voice solutions were used during this event. The Analog system GITI created was favored by the experiment participants due to issues found with the PBX. Issues identified with the PBX system were with adding users to conferences, creating delays and the choppiness of voice quality. The goal of this exercise was to continue the greybeard design helix development cycle. In addition we began integrating a software based voice system. We implemented a system that was client sever based with an onboard soft phone. We learned that the Conference Bridge in the software based voice system was not adequate to provide us with quality intercom loops. We continued to use old analog system in conjunction and replacement and continued to investigate VOIP as the long term solution. We learned that the Voice system we were using did not adequately meet our needs and was unreliable at best, we needed to continue looking for an alternative.

3.5.4 CommandEx 4

GITI supported this event held at ISX in April 2003. The setup included the addition of 2 controller stations and 4 more player stations, making a total of X users in the exercise. This was the first experiment with 6 Controllers. The PBX based voice solution was given another performance opportunity with greater integration of the MAYA software. At this event, it was decided to forego the PBX based solution and begin implementation of a mobile VOIP solution. There were too many issues with the ability to conference on the PBX hardware and the lack of portability of the system. In the late game of the original CPOF funding we were trying to gain some attention at this point. Our goal of this exercise was to demonstrate our quality product to the observing world. We continued to work in a Helix environment with greybeards and developers side by side working on the simulation and the design of features at the same time. We drew significant Army attention in these exercises. We learned in this exercise that although we had come a long way in the development of this program, we needed to make significant improvements in the implementation before we were ready to deploy this system into the hands of our soldiers.

3.6 Technology Transition Demonstrations

GITI supported multiple iterations of the CPOF technology at ISX in Arlington from July through September 2003. Although the standard exercises had become smaller, (more like demonstrations), each workstation was configured with dual CPUs and three or four displays. Some machines also had touch screens to simplify the input of hand drawings.

GITI attended and supported planning meetings for the CPOF Marine-Ex held at Quantico, Va. This included site survey and the coordination of network, computer hardware, furniture and facility support for the exercise.

GITI configured, installed, supported and de-installed the equipment to support the CPOF exercise at Quantico held in the Marine Corps Warfighting Lab (MCWL).

GITI participated in planning meetings for deployment of CPOF technology to the 1st Calvary, US Army for transition. This included multiple phone teleconferences and meetings held both at ISX in Arlington and at the Battle Command Training Center (BCTC) in Ft. Hood, Texas. GITI purchased and configured additional computing resources for Marine-Ex at Quantico and for the Army transition exercises at Ft. Hood. GITI transported CPOF Equipment to Ft. Hood, TX for use at the Battle Training Command Center for training CPOF to the green suits of the 1st Calvary. GITI installed and configured Gigabit and 100 MB networks in addition to analog and VOIP voice solutions at the BCTC at FT. Hood for training exercises for the 1st Calvary. GITI continued to administer and support the CPOF ftp site for coordination and distribution of data, maps, builds and presentations.

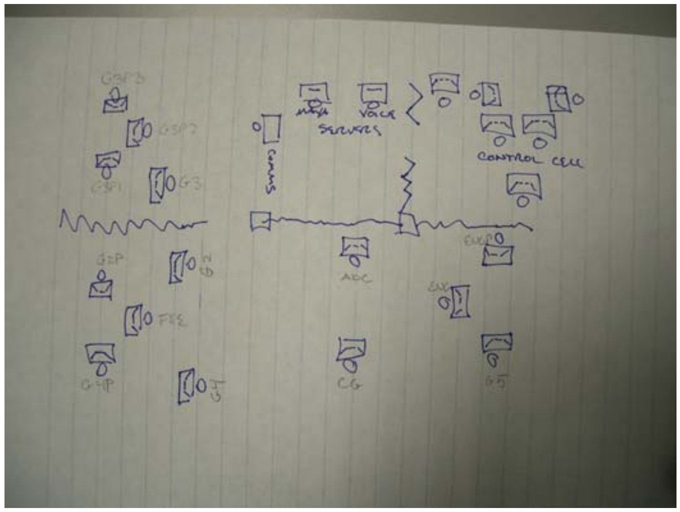


Figure 3.15 - Player station Diagram from BCTC Ft. Hood TX

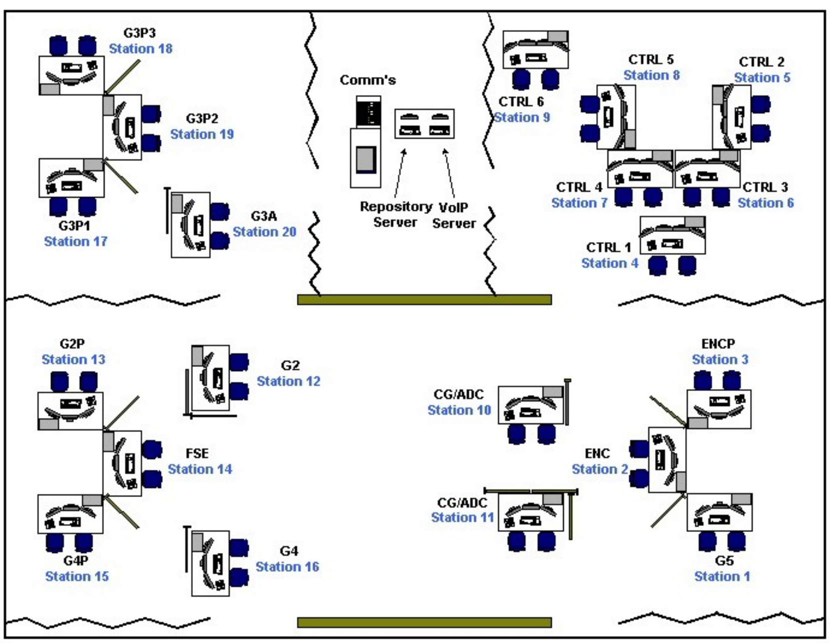


Figure 3.16 – BCTC Ft. Hood TX Position Diagram

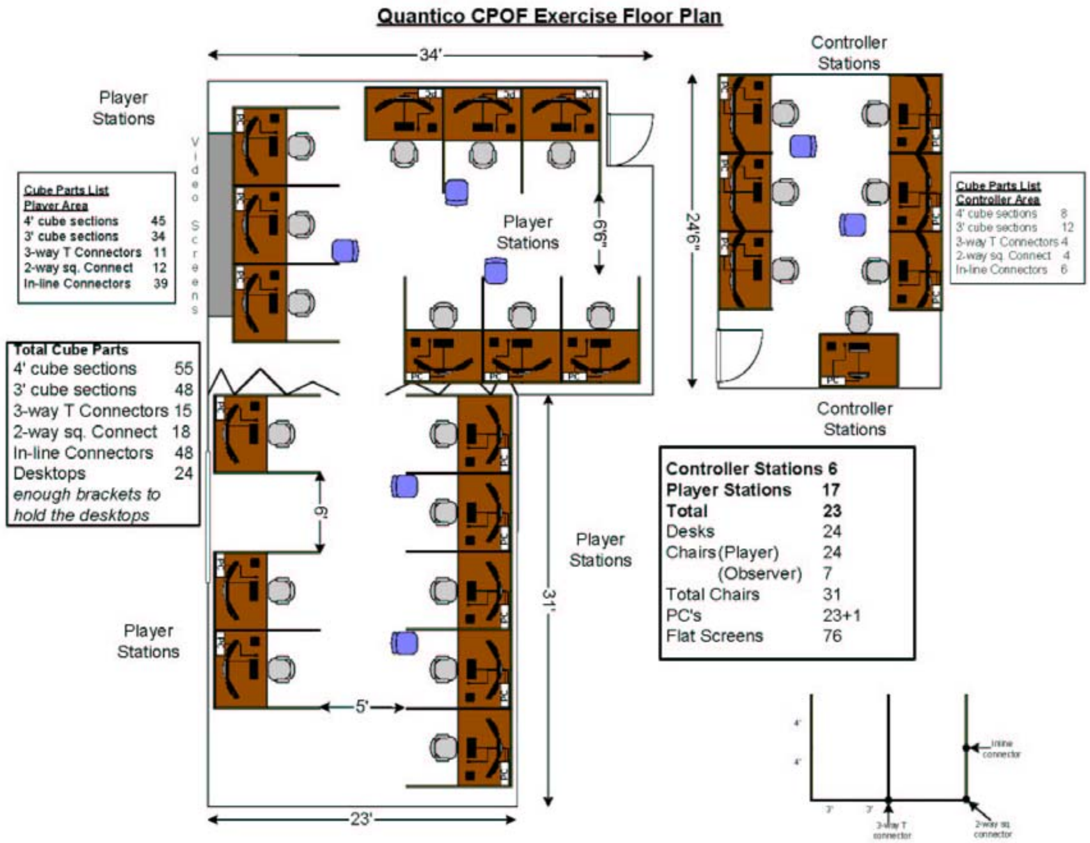


Figure 3.17 - MRX Exercise Layout Ft. Hood TX

4. Modeling and Simulation – CPOF Visualization

4.1 CMDR Overview

Imagine having the capability to transform any application dynamically, at run-time, into one capable of publishing and subscribing its data over a network. Systems and data models that are available throughout the enterprise could dynamically advertise their services. Agent simulations will then be able to search among all advertised capabilities to match services that meet their unique mission requirements. This capability exists today using DARPA's CoABS Grid and Global InfoTek's **C**urrent **M**ission **D**ata via the **R**TI (CMDR).

4.2 Proof-of-concept CPOF Prototype

One approach toward evolving CPOF visualization components was to build a specialized, test harness to distribute mission data to the visualization components across the Internet. The problem of distributing mission data, however, is a problem that has been addressed in depth already by the DOD modeling and simulation community culminating in the specification of the Defense Modeling and Simulation Office's High Level Architecture (HLA) and

implementation of the Run Time Infrastructure (RTI). The HLA focuses on enabling system interoperability across distributed heterogeneous simulation applications. While the value of applying the HLA RTI to the problem of stimulating CPOF visualization is clear, the system engineering team is developing an RTI-LEIF (Lightweight Extensible Information Framework) federate (RTI enabled component) that will enable LEIF applications to interact with any simulation application over the RTI without knowledge of the RTI.

An initial prototype experiment has been conducted using LEIF 4.0 and the RTI. The demonstration showed two LEIF applications on different hosts exchanging position data via the RTI. Another goal is to enable LEIF-based applications such as the CPOF Visualization Components to interact with the CoABS Grid regarding mission data publish and subscribe requirements. This approach involves downloading a Java archive file (JAR) containing the appropriate RTI enabled LEIF classes.

CMDR software has been enhanced significantly in the last year to meet the special requirements of the CPOF projects and also incorporate lessons learned from using prior versions of the software.

4.3 CMDR Capabilities

CMDR is a powerful federation monitoring tool which can simultaneously access multiple federations from a single application. It is an HLA/RTI Federation recorder which can capture the event stream of any Federation and, at another time or even location, playback such recordings with 100% fidelity. CMDR provides a convenient high level abstraction of the RTI library. Applications built on CMDR need no special knowledge of any particular RTI version. CMDR has a built in **F**ederation **O**bject **M**odel (FOM) **parser** with runtime bindings which automatically convert any FOM into a data schema and any data defined by that FOM into Java Objects. CMDR is both a federate **Publisher** and **Subscriber** providing a bi-directional communications channel which transparently exchanges federation data with other environments.

CMDR is a general purpose tool for rapid integration of HLA compliant federations with non-HLA components. CMDR provides a framework for interacting with the RTI libraries in an abstract manner to provide reusable applications with new federations by divorcing the application from low-level RTI structures and data formats. CMDR is the choice for expeditious interface of RTI Federate simulations or real-world data sources with other simulation modeling systems.

4.4 CMDR Applications

CMDR can be applied in five different configurations. CMDR has been used as a Composable Simulation Service in Global InfoTek's Composable Planner, as an HLA to CoABS Gateway in the EXMON Federation, a C4I to HLA/RTI simulation system, a library abstraction, and an HLA/RTI recorder/player.

HLA to CoABS Gateway
CMDR can be used to transparently interconnect CoABS Agents to any HLA Federation. In this application model, the CMDR executable acts as a proxy federate on behalf of one or more agents operating in the CoABS universe. These agents will see CMDR as the owner and source of all data originating in the federation. Similarly, the HLA Federates in the federation will see CMDR as the owner and source of all Entities, Attributes and Interactions which originate from the CoABS Grid Agents. CMDR does all the data caching and message routing necessary to preserve this illusion. The actual EXMON C4I-to-Simulation Federation made use of all of these services to seamlessly create a single simulation federation which was actually composed of a number of HLA Federates and several C4I Agents.

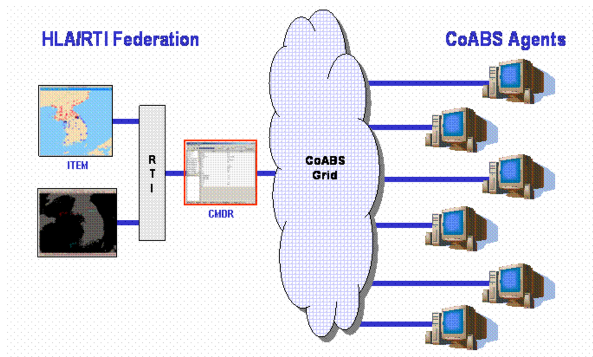


Figure 4.1 – HLA and Grid Gateway

In a manner analogous to CMDR's function in the composable simulation environment described in 2.2, CMDR can publish a web friendly version of the FOM describing the federation data model for lookup and examination by any client agent able to access the CoABS Lookup Service. Using the XML Schema describing the federation data model, any web based agent with access to the CoABS Grid communication layer can successfully interact with the Federation.

4.5 CMDR as a Composable Service
CMDR need not be invoked as a standalone application at all. By utilizing the **L**ook **U**p **S**ervices (LUS) provided by the Grid, a simulation composition application such as GITI's Composable Planner can make use of the Jini^{tm} service plug-in architecture to locate a headless CMDR instance running as a service, load its user interface and access the data it exports from its Federation seamlessly. The potential this has for the future of on-demand simulation composition or even real-time data reporting cannot be overestimated. CMDR puts the power of these legacy applications and services at the disposal of the users as if they were networked Grid agents or Jini Entries. As new agents or services are provided on the network, these can instantly benefit from legacy capability without having to wait for a new software deployment. New versions of existing tools can be discovered and utilized in the same manner.

4.6 Plug-in Architecture

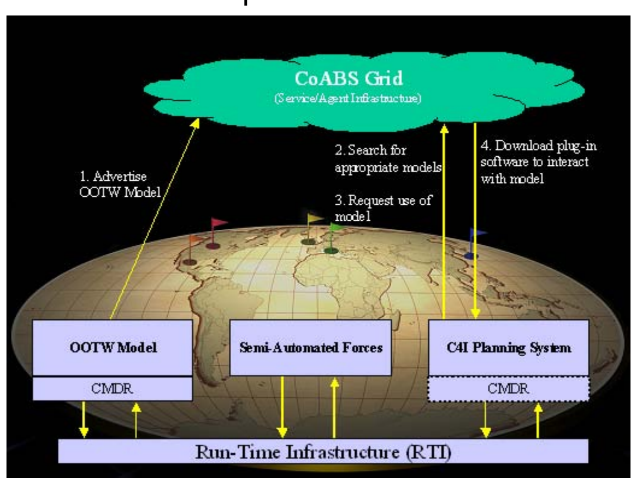
The most powerful feature of the Composable Planner is its ability to use software plug-ins to extend its basic capability. The architecture is designed for flexibility and reusability. The plug-ins can be provided from the local computer system or can be downloaded off the network and incorporated into the application. By using Java's introspection and reflection, the downloaded plug-in will be interrogated to determine the provided capabilities. It might be determined, for example, that the plug-in provides additional toolbar features. The plug-in architecture will add the newly discovered features to the user's toolbar. CMDR provides a plug-in module satisfying these criteria, making any data it exports readily available to the application.

4.7 Service User Interface

Through the CMDR plug-in, the architecture will incorporate the ability to become an HLA federate. CMDR will register with the Grid registry when it starts. Potential clients will use the LUS to locate available services which will include CMDR. CMDR's advertisement in the LUS will include the ontology of any FOM it presently proxies. CMDR also advertises its Service User Interface. Potential clients will see that CMDR has a service UI and they will also know what toolkits, such as AWT, Swing or SWT, are necessary to render it. CMDR is swing based, and any client with swing support can load and use CMDR's serialized UI Factory class to instantiate a CMDR UI and control its operation, thereby gaining access to the encapsulated HLA Federation.

4.8 Ontological Advertisement

A key function of CMDR is the ability to proxy Federations onto the Grid so that they may be discovered by other Agents and utilized. To accomplish this, the CMDR advertises itself as an available Grid Agent. Other CoABS Grid Agents will discover the Federation as an Advertised Agent. Jini based services will see the Federation as a Jini Entry. This advertisement consists of a description of the models capabilities, the elements of the simulation object model (SOM), and an XML Schema of the FOM CMDR is proxy for. This allows other agents, services, legacy systems, or applications to search the model's offered capabilities as represented in the LUS.



4.9 CMDR As Grid Agent

The CoABS Grid (Grid), developed at GITI under DARPA's CoABS program, arguably provides the most successful and widely used infrastructure to date for the large-scale integration of heterogeneous agent frameworks with object-based applications, and legacy systems. The CoABS Grid is an agent architecture designed to provide dynamic registration, search and discovery of agents and services. Illustrated in **figure 4.2**

is a C4I Planning system that is Grid-Aware, or capable of searching and viewing advertised agents and services. In this system, CMDR is deployed as Grid Agent.

CMDR is a fully enabled Grid Agent, it includes a method-based application-programming interface to register and advertise capabilities, discover services based on capabilities, and provide the necessary communication between services. In this application, CMDR wraps one or more legacy RTI components and presents them to the Grid as a surrogate Agent, giving the legacy application all the appearances and trappings of a genuine Agent. CMDR will translate its FOM and publish it in the LUS; find and work with other Agents of interest to the legacy application and adapt to changes in Agent environment such as component failure. CMDR takes advantage of Grid access to shared policies, ontologies, and services that support interoperability among agents and legacy simulations across a network infrastructure. For their part, the other Grid Agents and Services are completely unaware of CMDR's proxy function. It is entirely transparent, automatically publishing in both directions.

4.10 CMDR as Reusable Libraries

CMDR can be utilized as middleware between the application code and the RTI allows developers to rapidly develop core HLA compliant applications by providing a suite of convenient, easily understood abstractions of the RTI. This middleware is implemented only once, and then reused by each application. Reuse can be achieved by direct inheritance or aggregation from the CMDR classes or indirectly by way of a convenient CoABS grid Agency. **Figure 4.3** is a block diagram representing the layering of the CMDR Architecture.

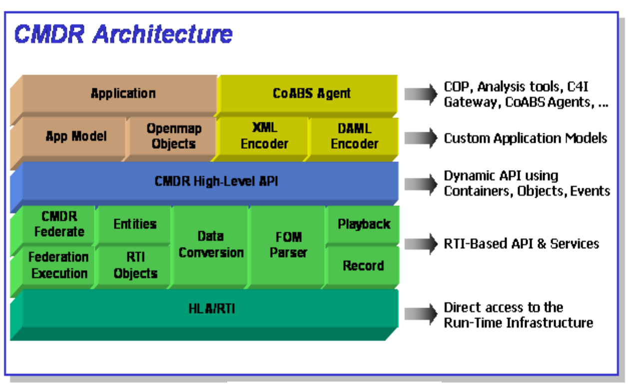


Figure 4.3 – CMDR Architecture

CMDR provides a number of reusable, high level application services. These are detailed as follows:

4.11 Data Caching Service

CMDR maintains a representation of all remote objects and distinguishes which domain owns them. New objects are added by the discoverObjectInstance RTI service, old objects removed by the removeObjectInstance RTI service, and object attributes are updated by the reflectAttributeValues RTI service. Similar operations are present in the Agent domain by agreement with the Agents for objects owned by that domain. Sometimes attributes are updated to their same value, for instance the heartbeat that indicates an object still exists appears as a complete update of the attributes. CMDR offers a feature to improve an application's performance by optionally filtering out updates that do not actually change a value thus reducing needless, expensive communication. CMDR automatically publishes attribute value updates into any domain other than that which originated the change. Interactions and transient events are similarly represented.

4.12 Agile FOM – A Data-driven Architecture

A major issue in the development of an HLA compliant simulation is the ability of a single federate to participate in multiple federations using different Federation Object Models (FOM). Current efforts to mitigate these problems through the use of standard names and formats, while important and necessary, do not solve the problem since the ability to use different representations is a powerful feature of the HLA. Object model independence was an important consideration when developing CMDR and was the reason an internal and flexible information model was chosen. Applications built with CMDR can be quickly adapted for new federations since CMDR uses the FOM to automatically learn about the data types available and map their conversion into objects. The framework implements the agile-FOM concept by allowing the application to work with new FOMs at any time simply by reloading the FOM or by instantiating a new CMDR with the desired FOM. This last means enables CMDR applications to access more than one federation simultaneously from one application.

4.13 Transparent Data Conversions

When the CMDR receives an incoming attribute update, it determines the proper converter to use for decoding and converting the update. This conversion happens automatically based on information in the FOM. Custom converters may supply FOM data types directly to application specific models by overriding the built in CMDR converters. The same process occurs in reverse for outgoing updates. This allows multiple conversions to be concatenated. For example, unit conversions between the FOM and the client applications internal representation can be implemented transparently. To facilitate this, CMDR represents all data types internally as Java Objects. It is then a simple matter to provide a conversion to any other encoding. CMDR presently supports RTI, Java Objects, Serialized Java Objects, XML and DAML representations of application data objects.

4.14 CMDR as HLA/RTI Data Recorder

CMDR provides unique capability to record and then playback at some later time, with complete fidelity, the data and interactions of an HLA Federation. In this

application, CMDR creates a Java object record of all Entities, Attribute Updates and Interactions which occur during the recording phase. This permits the use of HLA data off-line. The potential this provides for utilizing significant real-world data with simulations after the fact is enormous. For example, GCCS has been fitted with an RTI Federate proxy, Ambassador. [Ambassador is a communications middleware produced by the Naval Research Laboratories which permits DIICOE's **G**lobal **C**ommand **C**ontrol **S**ystem (GCCS) to appear as an HLA compliant RTI federate.] Ambassador and CMDR are together able to make such a recording for any scenario or situation for which GCCS has recorded a Reconstruction. The net effect allows simulators to replay and interact with a real-world engagement. **Figure 4.4** displays some of the possible configurations CMDR Record/Playback can be used in.

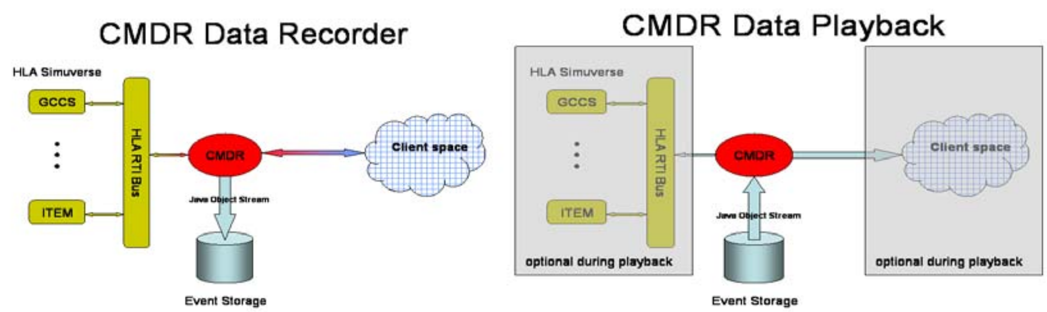


Figure 4.4 – CMDR Record/Playback Diagram

4.15 Recording Portability
Recorded HLA sequences are portable. They may be collected in one environment and later replayed in a different or even several environments. The potential for independent teams to share identical data and collaborate on the same project is obvious. This can be of great value in situations where one or more sub-teams are isolated by strict network security or geographical distance.

4.16 Enduring Freedom Reconstruction Activity
At the request of the CPOF PM, GITI attended an integration meeting in Burlington, Massachusetts for the Enduring Freedom Reconstruction Activity that simulated a key battle in Afghanistan. GITI restaged and presented the CMDR demo which showed how CMDR could integrate XIS and a number of simulation platforms using the RTI. GITI participated in two integration sessions which resulted in a video tape ultimately viewable by IDA and others involved with EFR. The tape shows how data coming in from the simulations can be put into XIS using CMDR, and how CMDR will keep XIS entities in sync with the simulations.

4.17 CMDR Summary and Lessons Learned
Global InfoTek, Inc has successfully developed a number of applications using the CMDR framework. GITI originally developed this capability to support requirements for the Command Post of the Future where MODSAF and JSAF

data needed availability on third party visualization systems. GITI integrated CMDR with the CoABS Grid to offer C4I planners the ability to utilize the services of distributed simulation and modeling systems while planning within traditional C2 systems. GITI used the improved framework in support of the Enduring Freedom Reconstruction (EFR) activity.

Our work with CMDR and the simulation community exposed a need to bring multiple HLA/RTI federations together. GITI typically refers to this as “A Federation of Federations”. Traditional HLA systems require each participant to process data using a specific Federation Object Model (FOM). Historically, this has proven to be difficult. Many simulations are hard-coded to a particular object model. Supporting a different FOM, or even changing the current one, can require many hours of development work. To avoid this complexity, GITI demonstrated that CMDR can accurately transform an HLA/RTI Federation into a CoABS service. The service acts as a bi-directional data pipe, publishing data from the RTI and pushing changes and interactions back in. By dynamically communicating with several of these federation services, an application can easily participate in multiple disparate federations at the same time.

The need for a capability to record and accurately playback data from an RTI federation was also exposed during our work on EFR. CMDR was instrumented with a rudimentary capability, which allowed several developers to work with recorded simulation data feeds even though they were not in direct contact with the simulation systems. This resulted a tremendous reduction in both development and integration time.

4.18 Proposed Future Enhancement

Global InfoTek would like to continue the development of CMDR. In particular, we see room for improvement in the following areas:

- CMDR could be enhanced to graphically map data between multiple FOMS. This is currently done with an XML configuration file, but end users should be empowered to do it through a convenient user interface.
- The CMDR record and playback mechanisms need to allow for editing of the event stream, insertion of playback pauses, and the concatenation of multiple streams.
- CMDR can be extended to include automated conversion of complex RTI data structures into standard Java classes. This would remove the need for custom converter classes and further reduce the work needed to integrate data from multiple federations.
- GITI would like to explore using CMDR with non-DMSO implementations of the RTI spec. The PITCH RTI and the IEEE spec interfaces are of particular interest. Currently, CMDR only supports RTI 1.3NGv4-6.

5. Flocking Algorithm

On 28 April 2000, the CPOF Program Manager requested GITI to assess the utility of flocking algorithms to support CPOF visualization. GITI designed and implemented flocking algorithms to model troop movements using LEIF mapping and information management software. GITI demonstrated the following capabilities to the Program Manager:

- Natural movement over terrain and around obstacles by using low level vector forces to influence the velocity and position of each unit.
- Units follow a user designed path of waypoints in a set formation or in random flocking.
- When units encounter an obstacle, (terrain feature or hostile forces), the "flock" will circumvent the hazard. Units pick a path that will keep them at least a certain distance away, while at the same time following the shortest path to the goal.
- Troops follow the squad leader
- Squad leader follows a user defined path
- Calculations are based on position and velocity vectors which are influenced by the goals of each troop Formations can be setup and modified through a text file
- Flocking algorithm can be demonstrated with troops moving in formation or in the more classic flocking movement
- Allows 3D positioning and movement in space.
- Virtually all the internal parameters affecting the underlying algorithm can be adjusted for different situations.

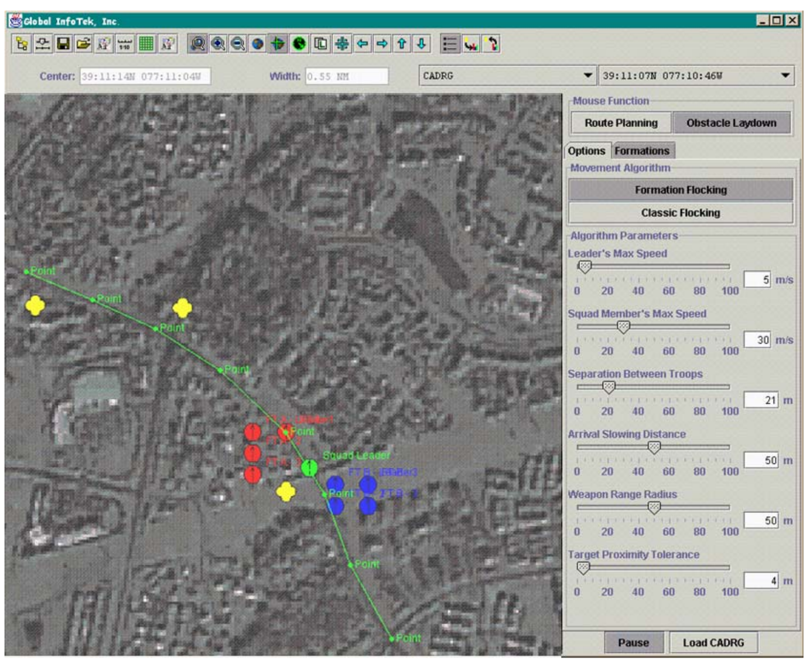


Figure 5.1 – Flocking Demonstration with XIS

6. Battle Authoring Tool

In January 2001, the CPOF Program Manager, requested that GITI assist Cycorp in integrating their Cyc reasoning engine with the MAYAViz software. The objective of this task was to assess the feasibility of using their reasoning system to support a real-time military command and control system. Due to changes in project priorities and the absence of an API for MayaViz software, the CPOF Program Manager directed GITI to evaluate the use of Cyc in a standalone environment using a GITI-developed Battle Authoring Tool to simulate battlespace events that the Cyc engine could reason upon.

GITI worked closely with Subject Matter Experts who were retired senior military officers to ensure development of a Battle Authoring Tool that could emulate and support realistic military operations. A key objective of our design was to develop an extremely friendly system for the SMEs to readily create battle scenarios.

BAT is a map-based digital system that allows the creation, modification, and deletion of units, reports, and events over time. It enabled CPOF SME's to quickly create reports to be sent to a Cycorp inference engine tasked to identify enemy reconnaissance and massing of artillery.

BAT was designed to allow an author to place "actors" (units, events, reports) on a "stage" (map) and direct them over time (game clock and event list). Authors may re-play the action using the playback tool.

The overall architecture of the Battle Authoring Tool is depicted in the following figure.

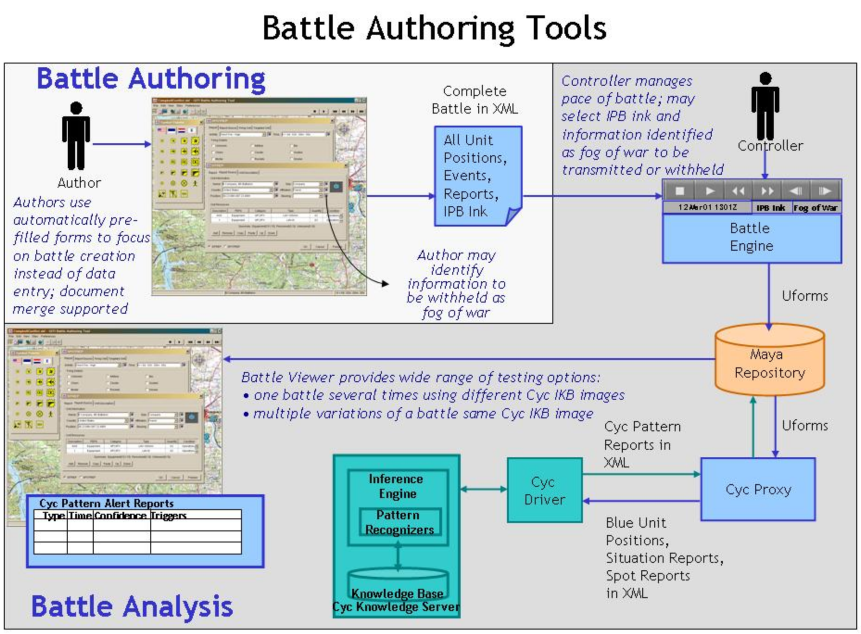


Figure 6.1 – Battle Authoring Tools

Using the Battle Authoring Tool, users create what is effectively a stage script for a battle. Units take the place of theatrical characters, and stage directions are comprised of maneuvers and various types of force attrition. Spoken words are replaced by SITREPs, SPOTREPs, and GPS Reports.

The battle author first sets the battle clock to a start time (known as H-hour). By creating units at this point, the user effectively declares the cast for his battle. The user can easily set the clock to any time before or after H-hour and create events such as the relocation of a unit, a change in unit status, or the creation of a particular report. By entering events that cover the timeline of a real battle, a user can create an accurate representation of the battlefield which can be played forward and back as if it were recorded on a VCR.

The BAT can also be used as a basic playback tool. By serving as the user interface for the Battle Engine, it offers the analyst an ability to play a battle up to a certain point and pause it. Playback can be continued with a simple click of the mouse. At any point in the battle, an analyst can double-click a unit to view its current state. Available data includes speed, direction, resources, and damage.

The Battle Authoring Tool uses a pluggable architecture, which allows external software to be registered as a consumer of battle data. Once plugged into the Battle Engine, third-party applications such as the Maya Repository and the CyC Inference Engine are fed all the Battle Events as they are played back. Any of these packages can then push alerts back into the BAT itself. In practice, this allowed CPOF to test technologies which were designed for real-time battle analysis. It also provides the necessary hooks to build software to augment the capability of the analyst in the future.

6.1 Battle Authoring Tool Features

Key features of the BAT are:

- Symbol Palette
  - Resource list templates
  - Default duration events
- Event List
  - Displays time, event type, unit involved
- Unit Editor
  - PEPA resource list
- Spot and Situation Reports
  - Wide range of activities
- Unit Table
  - Includes units such as JSTARS
- Report Table
  - Summary of reports
- Battle Playback Tool
  - Play

- Step to the next event
- Step to the previous event
- Pause
    - Fast forward to the end
    - Backward to the beginning

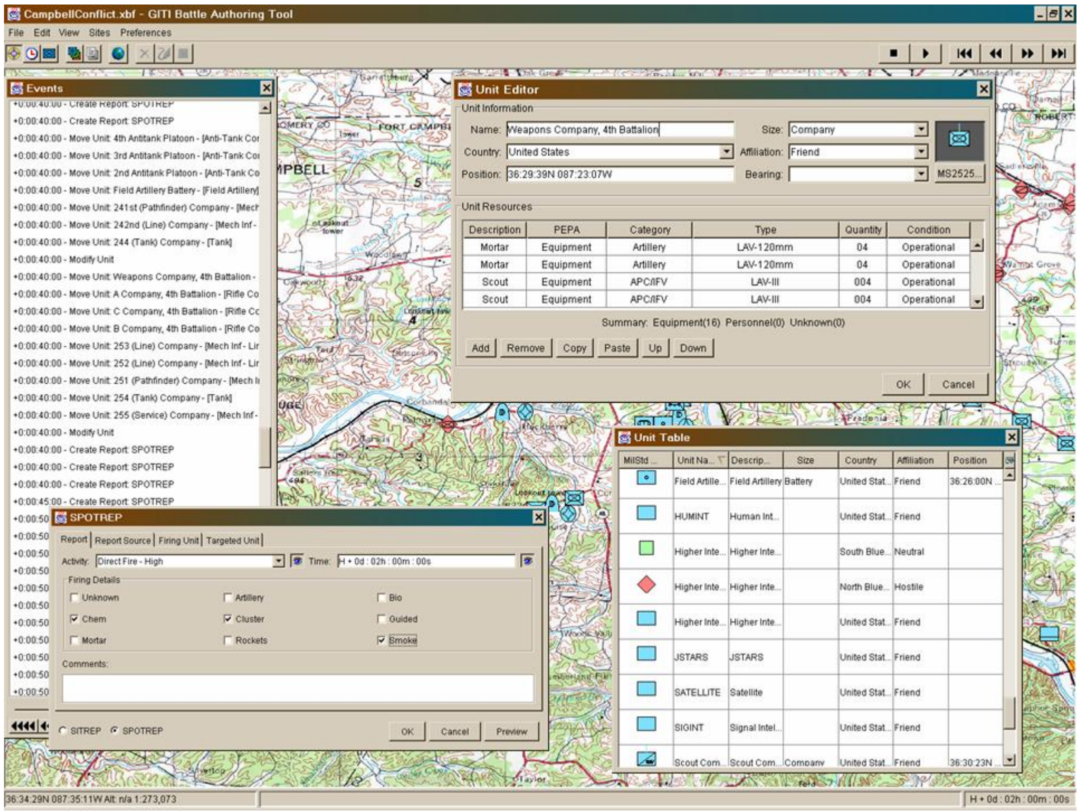


Figure 6.2 – Battle Authoring Tool User Interface

6.2 Battle Authoring Tool Assessment

A group of 5 authors created 800 reports in 1 ½ days. One Subject Matter Expert describes his recent experience using the BAT as follows:

“This second session gave me a much deeper appreciation of the richness and flexibility of the GITI Battle Authoring Tool (BAT). In particular, I found myself able to model all sorts of near future systems - fiber optic guided missiles (like the ones being used these days to assassinate Palestinian leaders), UAVs, long-range sensors of various kinds, and the like.

I particularly liked the ability to lay out a basic battle and then add “layers” of detail. So, after laying out the basic movement of units, I added a layer of JSTARS/higher intelligence reports, then a layer of indirect fire battles, and then some direct fire battles, and then an electronic warfare battle.

This experience has made me a big fan of the BAT, not merely as a means of providing battles for the Cyc engine to make sense of, but also as a tool for military education, war gaming, and combat development. I found, for example, that the act of authoring this battle gave me a very good sense of what was useful in the Army's organizational design for the medium brigade and what needed to be fixed."

6.3 Summary of Battle Authoring Tool and Lessons Learned

Battle Authoring tool was successfully used by the SMEs. This tool stimulated Cyc's inference engine. This experiment indicated that the Cyc inference engine that was used for the evaluation required significant computing power and could not currently support the requirements for near real-time command and control system.

Based on our interactions with SMEs, we have found out that typical military modeling and simulations, such as JSAF, are too complex and rigid for their use. Hence, there is a need for a next generation of tools similar to BAT that can be used by SMEs to create scenarios. Ideally the scenarios generated using these tools can be exported to simulation systems.

The state of the art inference engines are extremely computationally intensive. These systems cannot be used as a general purpose inference engine to reason about all events. However, we found that reasonable results can be obtained when the problem space is sufficiently small and highly focused on a specific set of events.

6.4 Proposed Future Enhancement

- Develop visual representations for Cyc phase 2 -- pattern interpretations
- Replace MIL-STD 2525 with alternative tactical symbols
- Add weather report
- Provide unit drill down/roll up
  - Select Battalion icon and expand to display Companies; select Company icon and expand to display Platoons
- Display battle effectiveness
  - Display fan of unit battle effectiveness & vulnerabilities based on current unit capabilities, readiness, and terrain
- Provide collaborative authoring
  - Allow multiple authors to create a single battle
- Provide collaborative battle exercise
  - Allow multi-player game play